

**Amendments to the Claims:**

This listing of claims will replace all prior versions and listings of claims in the application.

**Listing of Claims:**

1. (Currently Amended) A method for modifying a dynamic table within an array while maintaining integrity of subsequent tables in the array, the method comprising:
  - editing the dynamic table by selectively inserting at least one record to or deleting at least one record from the dynamic table, wherein the at least one record does not encompass an entire row of the array;
  - determining whether a subsequent table within the array includes ~~a first~~ an initially unaltered first portion and a second portion that is shifted within the array by the step of editing the dynamic table; and
  - shifting the first portion of the subsequent table within the array to align the first and second portions of the subsequent table.
2. (Previously Presented) A method as defined in claim 1 wherein the array comprises a two-dimensional spreadsheet and each record comprises a separate row within the dynamic table, and wherein the step of determining whether a subsequent table includes a second shifted portion comprises:
  - calculating a Union Column Range (UCR) of cells that extends below the dynamic table; and
  - determining whether a column range of the subsequent table overlaps the column range UCR of the dynamic table, wherein the overlapping column range of the subsequent table comprises the second portion of the subsequent table.

3. (Previously Presented) A method as defined in claim 2 wherein the step of shifting the first portion of the subsequent table within the spreadsheet comprises:

calculating a collective Left Column Range/Right Column Range (LCR/RCR) of the subsequent table that extends outside of the UCR column range; and

shifting the LCR/RCR column range of the subsequent table to align the LCR/RCR column range with the second portion of the subsequent table.

4. (Currently Amended) A method as defined in claim 3 further comprising:

determining whether a third table within the spreadsheet includes ~~a first~~ an initially unaltered first portion and a second portion that is shifted within the spreadsheet by at least one of the steps of editing the dynamic table and shifting the first portion of the subsequent table; and

shifting the first portion of the third table within the spreadsheet to align the first and second portions of the third table.

5. (Original) A method as defined in claim 3 wherein:

the step of editing the dynamic table comprises inserting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted downward  $n$  rows; and

the step of shifting the first portion of the subsequent table within the spreadsheet comprises inserting  $n$  blank rows above the LCR/RCR column range of the subsequent table.

6. (Currently Amended) A method as defined in claim 5 further comprising:

(a) calculating a new column range UCR2 of cells that extends below the modified table and the subsequent table;

(b) determining whether a third table within the spreadsheet includes ~~a first~~ an initially unaltered first portion and a second portion that overlaps the column range UCR2; and

(c) inserting  $n$  blank rows above the first ~~unaltered~~ portion of the third table to align the first and second portions of the third table.

7. (Currently Amended) A method as defined in claim 6 further comprising:  
repeating steps (a), (b) and (c) for each table below the third table in the spreadsheet so that ~~a first~~ an initially unaltered first portion of each such lower table is shifted downward by  $n$  rows to maintain the integrity of all the lower tables within the spreadsheet.

8. (Original) A method as defined in claim 3 wherein:  
the step of editing the dynamic table comprises deleting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted upward  $n$  rows; and  
the step of shifting the first portion of the subsequent table within the spreadsheet comprises deleting  $n$  blank rows above the LCR/RCR column range of the subsequent table.

9. (Previously Presented) A method as defined in claim 3 wherein the step of editing the dynamic table comprises deleting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted upward  $n$  rows, and wherein the step of shifting the first portion of the subsequent table within the spreadsheet comprises:

calculating a maximum upward shift  $m$  for the LCR/RCR column range of the subsequent table, where  $m < n$  and the value of  $m$  is calculated to prevent data cells above the subsequent table from being deleted;

inserting  $(n-m)$  blank rows below the modified table in the UCR column range;

and

deleting  $m$  blank rows above the LCR/RCR column range of the subsequent table.

10. (Currently Amended) A method as defined in claim 3 wherein the step of editing the dynamic table comprises deleting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted upward  $n$  rows, and wherein the step of shifting the first portion of the subsequent table within the spreadsheet comprises:

(a) calculating a maximum upward shift  $m_1$  for the LCR/RCR column range of the subsequent table, where  $m_1 < n$  and the value of  $m_1$  is calculated to prevent data cells above the subsequent table from being deleted;

(b) calculating a new column range UCR2 of cells that extends below the modified table and the subsequent table;

(c) determining whether a third table within the spreadsheet includes ~~a first~~ an initially unaltered first portion and a second portion that overlaps the column range UCR2;

(d) calculating a maximum upward shift  $m_2$  for the LCR/RCR column range of the third table, where  $m_2 < n$  and the value of  $m_2$  is calculated to prevent data cells above the third table from being deleted;

(e) selecting a minimum shift value  $m_{min}$  that equals the smaller of the values for  $m_1$  and  $m_2$ ;

(f) inserting  $(n-m_{min})$  blank rows below the modified table in the UCR column range; and

(g) deleting  $m_{min}$  blank rows above the LCR/RCR column ranges of the subsequent and third tables.

11. (Original) A method as defined in claim 10 further comprising:

repeating steps (b), (c) and (d) for each table below the third table in the spreadsheet so that the value  $m_{min}$  selected in step (e) equals the smallest of all the values  $m$  for each of the tables below the modified table, and wherein step (g) further comprises deleting  $m_{min}$  blank rows above the LCR/RCR column ranges of all of the tables below the modified table.

12. (Original) A computer program product readable by a computer and encoding instructions for executing the method recited in claim 1.

13. (Original) A computer program product readable by a computer and encoding instructions for executing the method recited in claim 5.

14. (Original) A computer program product readable by a computer and encoding instructions for executing the method recited in claim 9.

15. (Currently Amended) A system for modifying a dynamic table within an array while maintaining integrity of subsequent tables in the array, the system and the array stored in memory of a computer, the system comprising:

an input module for receiving edits to the dynamic table;

a table edit module for selectively inserting at least one record to or deleting at least one record from the dynamic table in accordance with instructions received from the input module, wherein the at least one record does not encompass an entire row of the array; and

a spreadsheet integration module for determining whether a subsequent table within the array includes ~~a first~~ an initially unaltered first portion and a second portion that is shifted within the array in response to an insertion or deletion of records in the dynamic table by

the table edit module, the spreadsheet integration module further operating to shift the first portion of the subsequent table within the array to align the first and second portions of the subsequent table.

16. (Previously Presented) A system as defined in claim 15 wherein the array comprises a two-dimensional spreadsheet and each record comprises a separate row within the dynamic table, and wherein the spreadsheet integration module further operates to:

calculate a Union Column Range (UCR) of cells that extends below the dynamic table; and

determine whether a column range of the subsequent table overlaps the column range UCR of the dynamic table, wherein the overlapping column range of the subsequent table comprises the second portion of the subsequent table.

17. (Previously Presented) A system as defined in claim 16 wherein the spreadsheet integration module further operates to:

calculate a collective Left Column Range/Right Column Range (LCR/RCR) of the subsequent table that extends outside of the UCR column range; and

shift the LCR/RCR column range of the subsequent table to align the LCR/RCR column range with the second portion of the subsequent table.

18. (Currently Amended) A system as defined in claim 17 wherein the spreadsheet integration module further operates to:

determine whether a third table within the spreadsheet includes ~~a first~~ an initially unaltered first portion and a second portion that is shifted within the spreadsheet by at least one of the edits to the dynamic table and the shifting of the first portion of the subsequent table; and

shift the first portion of the third table within the spreadsheet to align the first and second portions of the third table.

19. (Original) A system as defined in claim 17 wherein:

the table edit module edits the dynamic table by inserting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted downward  $n$  rows; and

the spreadsheet integration module shifts the first portion of the subsequent table within the spreadsheet by inserting  $n$  blank rows above the LCR/RCR column range of the subsequent table.

20. (Currently Amended) A system as defined in claim 19 wherein the spreadsheet integration module further operates to:

(a) calculate a new column range UCR2 of cells that extends below the modified table and the subsequent table;

(b) determine whether a third table within the spreadsheet includes a ~~first~~ an initially unaltered first portion and a second portion that overlaps the column range UCR2; and

(c) insert  $n$  blank rows above the first ~~unaltered~~ portion of the third table to align the first and second portions of the third table.

21. (Currently Amended) A system as defined in claim 20 wherein the spreadsheet integration module further operates to:

repeat steps (a), (b) and (c) for each table below the third table in the spreadsheet so that a ~~first~~ an initially unaltered first portion of each such lower table is shifted downward by  $n$  rows to maintain the integrity of all the lower tables within the spreadsheet.

22. (Original) A system as defined in claim 17 wherein:

the table edit module edits the dynamic table by deleting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted upward  $n$  rows; and

the spreadsheet integration module shifts the first portion of the subsequent table within the spreadsheet by deleting  $n$  blank rows above the LCR/RCR column range of the subsequent table.

23. (Previously Presented) A system as defined in claim 17 wherein the table edit module edits the dynamic table by deleting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted upward  $n$  rows, and wherein the spreadsheet integration module operates to:

calculate a maximum upward shift  $m$  for the LCR/RCR column range of the subsequent table, where  $m < n$  and the value of  $m$  is calculated to prevent data cells above the subsequent table from being deleted;

insert  $(n-m)$  blank rows below the modified table in the UCR column range; and

delete  $m$  blank rows above the LCR/RCR column range of the subsequent table.

24. (Currently Amended) A system as defined in claim 17 wherein the table edit module edits the dynamic table by deleting  $n$  rows to create a modified table, where  $n$  is an integer greater than zero, and wherein all cells within the UCR column range below the modified table are shifted upward  $n$  rows, and wherein the spreadsheet integration module operates to:

(a) calculate a maximum upward shift  $m_l$  for the LCR/RCR column range of the subsequent table, where  $m_l < n$  and the value of  $m_l$  is calculated to prevent data cells above the subsequent table from being deleted;



(b) calculate a new column range UCR2 of cells that extends below the modified table and the subsequent table;

(c) determine whether a third table within the spreadsheet includes a ~~first~~ an initially unaltered first portion and a second portion that overlaps the column range UCR2;

(d) calculate a maximum upward shift  $m_2$  for the LCR/RCR column range of the third table, where  $m_2 < n$  and the value of  $m_2$  is calculated to prevent data cells above the third table from being deleted;

(e) select a minimum shift value  $m_{min}$  that equals the smaller of the values for  $m_1$  and  $m_2$ ;

(f) insert  $(n-m_{min})$  blank rows below the modified table in the UCR column range; and

(g) delete  $m_{min}$  blank rows above the LCR/RCR column ranges of the subsequent and third tables.

25. (Original) A system as defined in claim 24 wherein the spreadsheet integration module further operates to:

repeat steps (b), (c) and (d) for each table below the third table in the spreadsheet so that the value  $m_{min}$  selected in step (e) equals the smallest of all the values  $m$  for each of the tables below the modified table; and

delete  $m_{min}$  blank rows above the LCR/RCR column ranges of all of the tables below the modified table.